

# Bootloader testing in LAVA

Rémi Duraffort, Linaro Ltd.



# Who am I?

- Rémi Durauffort
- Senior Software Engineer at Linaro
- LAVA Architect
- OSS developer since 2007
  - VLC media player
  - v8 js engine
  - PRoot/CARE
  - LAVA, lavacli, meta-lava, DummySys, lavafed, ...



# Why testing bootloaders ?

# Why testing bootloaders?

- **Critical** component
  - Can prevent the main OS to boot
- Bootloader **security**
  - Privileged hardware access
  - Chain of trust (secure boot)
- **Bugs** and **regressions**
  - Like every software



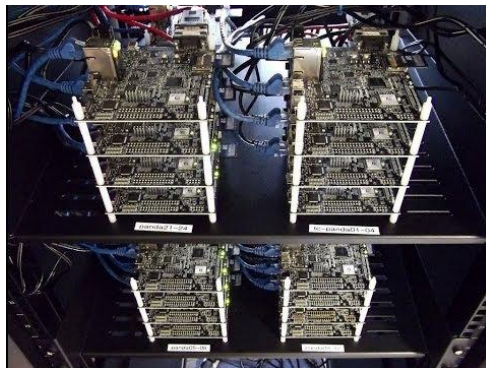
# Why testing bootloaders?

- Second/Third Stage Bootloaders are **complex**
  - Small OS (u-boot, LK)
  - Various protocols (fastboot, DFU...) and drivers (FS, SDIO, USB...)
  - Network stack (TCP/IP, TFTP)
  - New standardized interfaces (EBBR, UEFI...)
- Usually much **less tested**
  - Can include non-upstream code
    - vendor/oem customizations

LAVA

# LAVA

- Linaro **A**utomated **V**alidation **A**rchitecture
- Test execution system: **testing software on real hardware**
  - Boot testing: kernelci
  - System level testing: lkft
  - Power consumption
  - Benchmarks
  - Multinode
    - Test with many devices
  - ...



**LAVA**

[linaro.org/lava](http://linaro.org/lava)

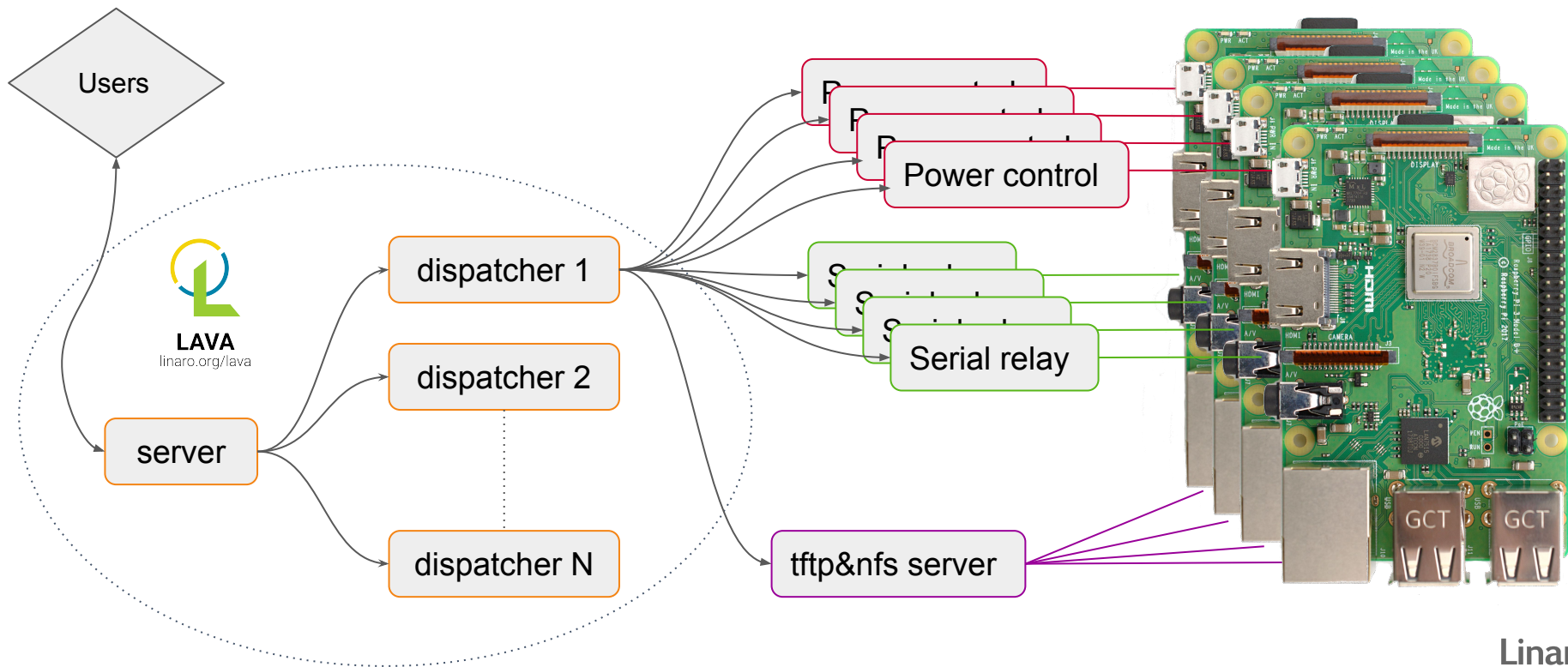
# LAVA

- Organized in a server-dispatcher model:
  - **lava-server**: master
  - **lava-dispatcher**: worker
- Distributed
  - Remote dispatchers
- Scalable





# LAVA architecture



# LAVA roles

## Server

- Web UI and API
  - Submit jobs
  - Results, logs, ...
- Access control
  - Users, groups
- Scheduling jobs
  - Priority
  - Multinode jobs
- Store job logs
- Send notifications

## Dispatcher

- Deploy resources
- Power on/off DUTs
- Parse logs
  - Kernel panic
  - Bootloader error
- Classify errors
  - Infrastructure, Bug
  - Job, Test, ...

# LAVA as a user

- Submit job definition
  - *lavacli jobs submit job-definition.yaml*
- Get results and logs

```
export SHELL=/bin/sh
root@rznl-snarc:/tmp# export SHELL=/bin/sh
export SHELL=/bin/sh
. /lava-259851/environment
root@rznl-snarc:/tmp# . /lava-259851/environment
. /lava-259851/environment
/lava-259851/bin/lava-test-runner /lava-259851/0
root@rznl-snarc:/tmp# /lava-259851/bin/lava-test-runner /lava-259851/0
Test shell timeout: 10s (minimum of the action and connection timeout)
/lava-259851/bin/lava-test-runner /lava-259851/0
+ export TESTRUN_ID=0_smoke-tests
+ TESTRUN_ID=0_smoke-tests
+ cd /lava-259851/0/tests/0_smoke-tests
++ cat uuid
+ UUID=259851_1.2.2.4.1
+ set +x
<LAVA_SIGNAL_STARTRUN 0_smoke-tests 259851_1.2.2.4.1>
+ lava-test-case linux-posix-pwd --shell pwd
<LAVA_SIGNAL_STARTTC linux-posix-pwd>
/lava-259851/0/tests/0_smoke-tests
<LAVA_SIGNAL_ENDTC linux-posix-pwd>
<LAVA_SIGNAL_TESTCASE TEST_CASE_ID=linux-posix-pwd RESULT=pass>
Received signal: <STARTRUN> 0_smoke-tests 259851_1.2.2.4.1
Starting test lava.0_smoke-tests (259851_1.2.2.4.1)
Skipping test definition patterns.
Received signal: <STARTTC> linux-posix-pwd
Received signal: <ENDTC> linux-posix-pwd
Received signal: <TESTCASE> TEST_CASE_ID=linux-posix-pwd RESULT=pass

case: linux-posix-pwd
case_id: 6800171
definition: 0_smoke-tests
result: pass
```

# RPi3 health-check

device\_type: "rpi3-b-32"

job\_name: "rpi3-b-32 standard health check"

timeouts:

  job:

    minutes: 10

priority: "medium"

visibility: "public"

actions:

- deploy:

  timeout:

    minutes: 6

  to: "tftp"

  kernel:

    url: "http://.../r0-raspberrypi3.bin"

    type: "uimage"

  modules:

    url: "http://.../raspberrypi3.tgz"

    compression: "gz"

  nfsrootfs:

    url: "http://.../rootfs.tar.xz"

    compression: "xz"

  dtb:

    url: "http://.../bcm2710-rpi-3-b.dtb"

# RPi3 health-check

## - boot:

```
timeout:  
  minutes: 4
```

```
method: "u-boot"
```

```
commands: "nfs"
```

```
auto_login:
```

```
  login_prompt: "login:"
```

```
  username: "root"
```

```
prompts:
```

```
- "root@raspberrypi3:~#"
```

## - test:

```
timeout:  
  minutes: 5
```

```
definitions:
```

```
- repository: "http://.../tests.git"  
  from: "git"
```

```
  path: "smoke-tests-basic.yaml"  
  name: "smoke-tests"
```

```
- repository: "http://.../qa.git"
```

```
  from: "git"
```

```
  path: "ltp.yaml"
```

```
  name: "ltp"
```

# Supported methods

## deploy:

- tftp
- nbd
- flasher
- fastboot
- mps
- ssh
- docker
- vexpress ums
- ...

## boot:

- cmsis
- dfu
- uboot
- pyocd
- fastboot
- docker
- qemu
- grub
- iso installer
- ...

## test:

- git repository
- interactive
- minimal
- multinode

# Supported device-types: 202

adb-nuc alpine-db am6 apq8016-sbc-uboot ar9331-dpt-module arduino101 armada-370-db armada-370-rd armada-3720-db  
armada-3720-espressobin armada-375-db armada-385-db-ap armada-388-clearfog armada-388-gp armada-398-db armada-7040-db  
armada-8040-db armada-xp-db armada-xp-gp armada-xp-linksys-mamba armada-xp-openblocks-ax3-4 arndale at91rm9200ek at91sam9261ek  
at91sam9m10g45ek at91sam9x25ek at91sam9x35ek at91-sama5d2\_xplained at91-sama5d4\_xplained b2120h410 b2260 base base.dpkg-dist  
bcm2836-rpi-2-b bcm2837-rpi-3-b-32 bcm2837-rpi-3-b beaglebone-black-barebox beaglebone-black beagle-xm cc13x2-launchpad cc3220SF  
cubietruck d02 d03 da850-lcdk disco-l475-iot1 docker dove-cubox dra7-evm dragonboard-410c dragonboard-820c dragonboard-845c  
exynos4-5-common exynos5420-common frdm-k64f frdm-kw41z fsl-imx8mm-evk fsl-imx8mq-evk hi6220-hikey-bl hi6220-hikey hi6220-hikey-r2  
hi960-hikey hifive-unleashed-a00 highbank hip07-d05 hsdk ifc6410 imx23-olinuxino imx27-phytec-phycard-s-rdk imx28-duckbill imx53-qsrb  
imx6dl-riotboard imx6q-nitrogen6x imx6q-sabrelite imx6ul-pico-hobbit imx7s-warp imx8m-common imx8m-jetson-tk1 jun0 jun0-uboot jun0-uefi  
kirkwood-db-88f6282 kirkwood-openblocks\_a7 kvm lava-slave-docker lxc mediatek-8173 meson8b-ec100 meson8b-odroidc1 meson-g12a-sei510  
meson-g12a-u200 meson-g12a-x96-max meson-g12b-a311d-khadas-vim3 meson-g12b-odroid-n2 meson-g12-common meson-gxbb-nanopi-k2  
meson-gxbb-p200 meson-gx-common meson-gxl-s805x-libretech-ac meson-gxl-s805x-p241 meson-gxl-s905d-p230 meson-gxl-s905x-khadas-vim  
meson-gxl-s905x-libretech-cc meson-gxl-s905x-p212 meson-gxm-khadas-vim2 meson-gxm-q200 mimxrt1050\_evk minnowboard-common  
minnowboard-max-E3825 minnowboard-turbot-E3826 moonshot-m400 mps mustang-grub-efi mustang-uefi nexus10 nexus4 nexus5x nexus9  
nrf52-nitrogen nucleo-l476rg nxp-ls2088 odroid-x2 odroid-xu3 orion5x-rd88f5182-nas overdrive ox820-cloudengines-pogoplug-series-3 panda peach-pi  
pixel poplar qcom-qdf2400 qcs404-evb-1k qcs404-evb-4k qemu-aarch64 qemu r8a7791-porter r8a7795-salvator-x r8a7796-m3ulcb rcar-gen3-common  
rk3288-rock2-square rk3288-veyron-jaq rk3328-rock64 rk3399-gru-kevin rk3399-puma-haikou rpi-common rzn1d sama5d3 sama5d34ek sama5d36ek  
sdm845-mtp sharkl2 snow soca9 socfpga-cyclone5-socrates ssh stm32-carbon stm32mp157c-dk2 sun4i-a10-olinuxino-lime sun50i-a64-bananapi-m64  
sun50i-a64-pine64-plus sun50i-h5-libretech-all-h3-cc sun50i-h6-orangepi-one-plus sun50i-h6-pine-h64 sun5i-a13-olinuxino-micro sun5i-gr8-chip-pro  
sun5i-r8-chip sun6i-a31-app4-evb1 sun7i-a20-cubieboard2 sun7i-a20-olinuxino-lime2 sun7i-a20-olinuxino-micro sun8i-a23-evb sun8i-a33-olinuxino  
sun8i-a33-sinlinx-sina33 sun8i-a83t-allwinner-h8homlet-v2 sun8i-a83t-bananapi-m3 sun8i-h2-plus-bananapi-m2-zero sun8i-h2-plus-libretech-all-h3-cc  
sun8i-h2-plus-orangepi-r1 sun8i-h2-plus-orangepi-zero sun8i-h3-libretech-all-h3-cc sun8i-h3-orangepi-pc sun8i-r40-bananapi-m2-ultra sunxi-common  
synquacer-acpi synquacer-dtb synquacer-tc2 tegra124-common tegra124-nyan-big thunderx upsquare vexpress x15-bl x15 x86-atom330 x86-celeron  
x86 x86-pentium4 x86-x5-z8350 xilinx-zcu102

# Bootloaders specificities



# Usual assumptions (OS testing)

- Deploying images
  - Known filesystem (ext4, fat, ...): inject test scripts
  - Provisioning protocol: fastboot, tftp, nfs, ...
  - Stable/unique identifier: USB ID, MAC
- Booting device
  - UART/console
  - Recovery method: for non functional OS
- Testing
  - POSIX OS/shell

**A working  
bootloader**

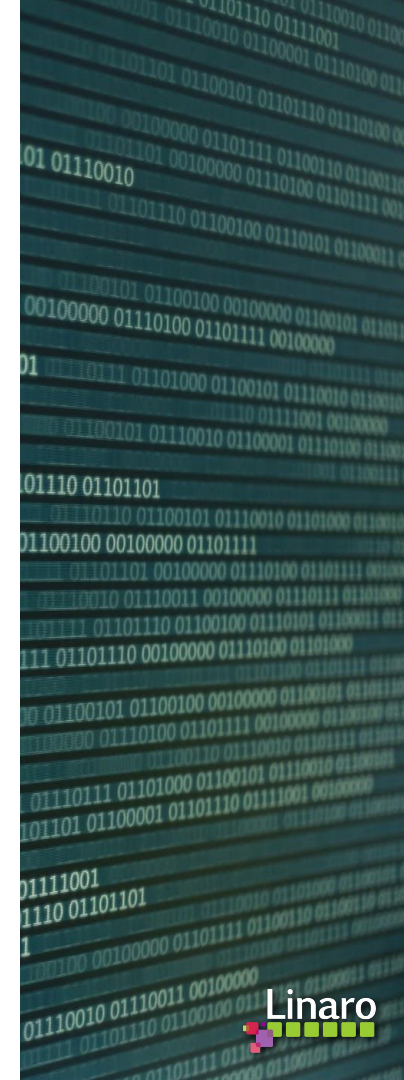
# Bootloader issues

- No POSIX shell/system
  - Can not run standard test scripts
- No filesystem
  - No easy way to inject tests (overlay) into tested system image
- Identifiers/Uniqueness
  - E.g. empty DFU serial number

# Bootloader issues

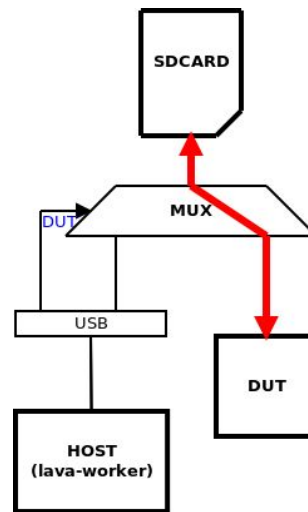
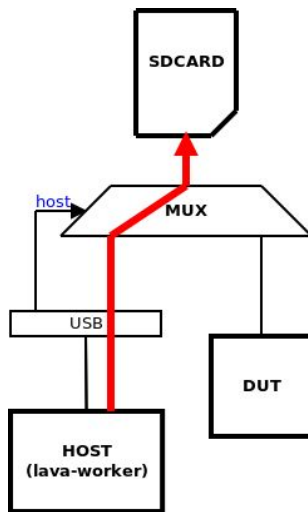
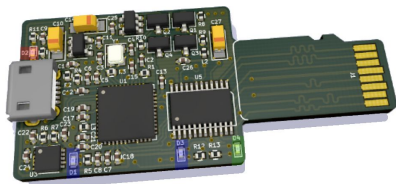
- Harder to deploy/recover
  - Usually responsible for system update
    - fastboot, TFTP, ...
  - Provision a new bootloader without relying on an operational one.
    - Recovery mode
    - Out-of-band provisioning

# Bootloader provisioning



# Provisioning with SDMux

- Multiplexing the SD-Card
  - either DUT or Host
- Remotely control the switch

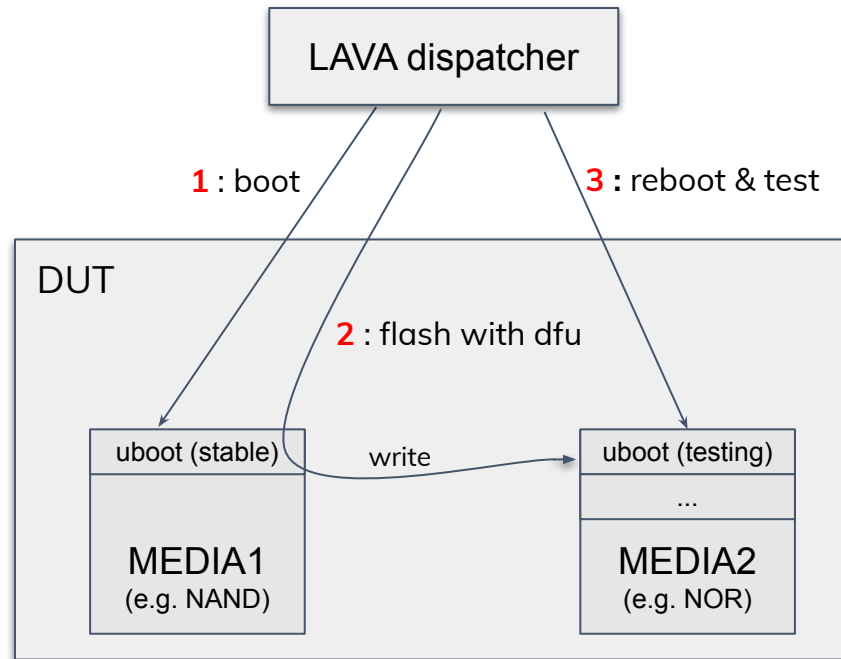


# Provisioning with SDMux

- Pros:
  - Don't rely on minimal functional system or recovery more
  - Full control of the memory layout
  - Simple setup
  - Open design
  - Reliable
- Cons:
  - Not working with every DUT/SDCard
  - More hardware to buy
    - More usb cables

# Provisioning without SDMux

- Depends on board features
- Two boot media
  - NAND: stable u-boot
  - NOR: Testing u-boot
- Remotely select boot media
- Boot on NAND
  - Flash NOR using dfu
  - Reboot
- Boot on NOR
  - Test bootloader
  - Boot to Linux



# Provisioning without SDMux

- Depends on board features
- Complex scenario
  - Set relay
  - Boot to u-boot
  - Activate dfu
  - Flash over df
  - Unset relay
  - Reboot
- User can still break it
  - Flashing the stable u-boot



# Interactive testing

# Interactive testing

- No POSIX shell/system
  - Can not run standard test scripts
- No filesystem
  - No easy way to inject tests
- **Only a serial console**

# Interactive testing

- Like an **expect** script
  - Strings to send
  - Expected results
    - Successes or failures
- LAVA will interpret the serial output
- Pros:
  - No need to install anything on the DUT
  - Reliable and versatile
- Cons:
  - No if/loop/...
  - Verbose

# Interactive testing

interactive:

```
- name: memory-test
  prompts: ["=> ", "/ # "]
  script:
    # set and test network interface
    - command: dhcp
      name: dhcp
      successes:
        - message: "client bound to address"
      failures:
        - message: "TIMEOUT"
          exception: InfrastructureError
          error: "dhcp failed"
    [...]
    - command: base
      name: default-base-address-offset
      successes:
        - message: "Base Address: 0x"
    [...]
```

```
- command: base B0000000
  name: set-address-offset-0xb0000000
  successes:
    - message: "Base Address: 0xb0000000"
```

```
# mw - memory write
# md - memory display
- command: mw 100000 aabbccdd 10
- command: md 100000 10
  name: mw-md-100000
  successes:
    - message: "aabbccdd"
```

```
# md - memory display
# cp - memory copy
- command: cp 100000 200000 10
- command: md 200000 10
  name: cp-md-200000
  successes:
    - message: "aabbccdd"
```

# Demo

# Future improvements

- Fetch scripts from git
  - Make versioning easier
  - Allow to share test scripts
- Allow to run a custom script
  - Inside a container
  - With access to the serial output
  - LAVA dispatcher only looking at the stream

Question?